**SAVEETHA SCHOOL OF ENGINEERING**

**SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES**

**COMPUTER SCIENCE AND ENGINEERING PROGRAMME**

**CSA02-C PROGRAMMING-QUESTION BANK**

**MODERATE QUESTIONS**

1. A year has 365 days but leap year consists of 366 days. This one day is added in the month of February. This month which generally has 28 days and also known as the shortest month in a year would get added with an extra day, which gives us a total of 29 days in that month. It is based on the Georgian calendar. Compile and Execute the C program for checking out whether a given year is a leap year or not.

Sample Input:

Enter Date : 04/11/1947

Sample Output:

Given year is Non Leap Year

Test cases:

1. 04/11/19.47
2. 11/15/1936
3. 31/45/1996
4. 64/09/1947
5. 00/00/2000

Source code:

#include<stdio.h>

main(){

int i,j,date,month,year,c=0;

float y;

char str[20],strdate[20],strmonth[20],stryear[20],temp[2],token[20];

printf("Enter date (dd/mm/yy): ");

scanf("%d/%d/%f", &date,&month,&y);

year=(int)y;

y=y-year;

if(date>0 || date>31){

printf("Invalid input...");

}

else if(month>0 || month>12){

printf("invalid input...");

}

else if(year>0 || y>0){

printf("invalid input...");

}

else{

if (year % 400 == 0) {

printf("%d is a leap year.", year);

}

else if (year % 100 == 0) {

printf("%d is not a leap year.", year);

}

else if (year % 4 == 0) {

printf("%d is a leap year.", year);

}

else {

printf("%d is not a leap year.", year);

}

}

}

1. Ask the user to enter any positive integer. Then divides the given number into individuals and adds those individual (Sum) digits. Compile and execute the c Program and display the output on the screen.

Sample Input:

Enter N value : 3

Enter 3 digit number: 143

Sample Output:

Sum of 3 digit number: 8

Test cases:

1. N = 2, 158
2. N = 3, 14
3. N = 4, 0148
4. N = 1, 0004
5. N = 4, 7263

#include<stdio.h>

int main()

{

int n,sum=0,m;

printf("Enter a number:");

scanf("%d",&n);

if(n>0){

while(n>0)

{

m=n%10;

sum=sum+m;

n=n/10;

}

printf("Sum =%d",sum);

}

else{

printf("Invalid input");

}

}

1. Permutation refers number of ways in which set members can be arranged or ordered in some fashion. Compile and execute the C program to print unique permutations of a given number.

Sample Input:

Given Number: 143

Sample Output:

Permutations are:

134

143

314

341

413

431

Test cases:

1. 0
2. 111
3. 505
4. -143
5. -598

#include <stdio.h>

#include <string.h>

void swap(char\* x, char\* y)

{

char temp;

temp = \*x;

\*x = \*y;

\*y = temp;

}

void permute(char\* a, int l, int r)

{

int i;

if (l == r)

printf("%s\n", a);

else {

for (i = l; i <= r; i++) {

swap((a + l), (a + i));

permute(a, l + 1, r);

swap((a + l), (a + i));

}

}

}

main()

{

char str[20];

printf("Enter a number : ");

scanf("%s",&str);

int n = strlen(str);

permute(str, 0, n - 1);

}

1. Write a program to print the all Odd numbers and number of even numbers in between M and N?

Sample Input:

M = 6

N = 15

Sample Output:

All Odd Numbers = 7,9,11,13

All Even Numbers = 8,10,12,14

Test cases:

1. M = 100, N = 100
2. M = 500, N = 100
3. M = -5, N = 4
4. M = 72, N = -72
5. M = 0, N = 0

#include<stdio.h>

void main() {

int num,m,n,i;

printf("Enter start range :");

scanf("%d",&n);

printf("Enter end range :");

scanf("%d",&m);

if(m>0 && n>0 && m>n){

printf("odd:\n");

for(i=n;i<m;i++){

if(i%2!=0){

printf("%d\n",i);

}

}

printf("even:\n");

for(i=n;i<m;i++){

if(i%2==0){

printf("%d\n",i);

}

}

}

else{

printf("invalid input");

}

}

1. Write a program to find the number of student users in the college, get the total users, staff users details from the client. Note for every 3 staff user there is one Non teaching staff user assigned by default.

Sample Input:

Total Users: 856

Staff Users: 126

Sample Output:

Student Users: 688

Test Cases:

1. Total User: 0
2. Total User: -143
3. Total User: 1026, Staff User: 1026
4. Total User: 450, Staff User: 540
5. Total User: 600, Staff User: 450

#include<stdio.h>

main() {

int num,t,n,i,nt,st;

printf("Enter the total number of user :");

scanf("%d",&n);

printf("Enter the number of teaching staff :");

scanf("%d",&t);

if(n>0 && t>0 && n>t+t/3){

nt=t/3;

st=n-(t+nt);

printf("Student users : %d",st);

}

else{

printf("Invalid input");

}

}

1. Write a program using function to calculate the simple interest. Suppose the customer is a senior citizen. He is being offered 12 percent rate of interest; for all other customers, the ROI is 10 percent.

Sample Input:

Enter the principal amount: 200000

Enter the no of years: 3

Is customer senior citizen (y/n): n

Sample Output:

Interest: 60000

Test Cases:

1. Principal: 2000 , Years: 0
2. Principal: 20000 , Years: -2
3. Principal: -2000 , Years: 2
4. Principal: 2 , Years: 2000
5. Principal: 0 , Years: 5

#include<stdio.h>

float interest(int P, int N)

{ char ch[1];

printf("Are you a senior citizen(y/n): ");

scanf("%s", &ch);

float SI,r;

if(strcmp("y",ch)==0){

r=12;

}

else if(strcmp("n",ch)==0){

r=10;

}

SI=P\*r\*N/100.0;

return SI;

}

main()

{

int p,n;

float r,Z;

printf("Enter Principal Amount : ");

scanf("%d",&p);

printf("Enter Time Period : ");

scanf("%d",&n);

if(p>0 && n>0){

Z=interest(p,n);

printf("\nSimple Interest : %.2f\n",Z);

}

else{

printf("Invalid input");

}

}

1. Write a program for matrix Multiplication?

Sample Input:

Mat1 = 1 2

5 3

Mat2 = 2 3

4 11

Sample Output:

Mat Mul = 10 25

1. 48

#include<stdio.h>

#include<stdlib.h>

int main(){

int a[10][10],b[10][10],mul[10][10],r,c,i,j,k;

system("cls");

printf("enter the number of row=");

scanf("%d",&r);

printf("enter the number of column=");

scanf("%d",&c);

printf("enter the first matrix element=\n");

for(i=0;i<r;i++)

{

for(j=0;j<c;j++)

{

scanf("%d",&a[i][j]);

}

}

printf("enter the second matrix element=\n");

for(i=0;i<r;i++)

{

for(j=0;j<c;j++)

{

scanf("%d",&b[i][j]);

}

}

printf("multiply of the matrix=\n");

for(i=0;i<r;i++)

{

for(j=0;j<c;j++)

{

mul[i][j]=0;

for(k=0;k<c;k++)

{

mul[i][j]+=a[i][k]\*b[k][j];

}

}

}

for(i=0;i<r;i++)

{

for(j=0;j<c;j++)

{

printf("%d\t",mul[i][j]);

}

printf("\n");

}

return 0;

}

1. Write a program to print the number of vowels in the given statement?

Sample Input:

Saveetha School of Engineering

Sample Output:

Number o vowels = 12

Test cases:

1. India is my country
2. All are my brothers and sisters
3. Why dry sky
4. Shy Try Cry
5. EDUCATION

#include<stdio.h>

#include<string.h>

#define MAX 100

main()

{

char str[MAX];

int a,vowel,consonant,i;

printf("Enter a string:");

scanf("%[^\n]s",&str);

vowel=0;

consonant=0;

a=strlen(str);

for(i=0; i<a; i++)

{

if(str[i] =='a' || str[i]=='e' || str[i]=='i' || str[i]=='o' || str[i]=='u' ||

str[i] =='A' || str[i]=='E' || str[i]=='I' || str[i]=='O' || str[i]=='U' )

vowel++;

}

printf("Total number of vowel = %d\n", vowel);

}

1. Write a program for matrix multiplication?

Sample Input:

Mat1 = 1 2

5 3

Mat2 = 2 3

4 1

Sample Output:

Mat Multiplication = 3 5

1. 4

#include<stdio.h>

#include<stdlib.h>

int main(){

int a[10][10],b[10][10],mul[10][10],r,c,i,j,k;

system("cls");

printf("enter the number of row=");

scanf("%d",&r);

printf("enter the number of column=");

scanf("%d",&c);

printf("enter the first matrix element=\n");

for(i=0;i<r;i++)

{

for(j=0;j<c;j++)

{

scanf("%d",&a[i][j]);

}

}

printf("enter the second matrix element=\n");

for(i=0;i<r;i++)

{

for(j=0;j<c;j++)

{

scanf("%d",&b[i][j]);

}

}

printf("multiply of the matrix=\n");

for(i=0;i<r;i++)

{

for(j=0;j<c;j++)

{

mul[i][j]=0;

for(k=0;k<c;k++)

{

mul[i][j]+=a[i][k]\*b[k][j];

}

}

}

for(i=0;i<r;i++)

{

for(j=0;j<c;j++)

{

printf("%d\t",mul[i][j]);

}

printf("\n");

}

return 0;

}

1. Write a program to find the sum and average of the elements in an array

Sample Input;

Array of elements = {16, 18, 27, 16, 23, 21, 19}

Sample Output:

Sum = 140

Average = 20

#include<stdio.h>

main( )

{

int a[25], n, i ;

float avg = 0, sum = 0 ;

printf(" Enter the Numbers of element in Array: ");

scanf("%d",&n) ;

printf("Enter an Element : \n") ;

for ( i = 0 ; i < n ; i++)

{

scanf("%d",& a[i]) ;

}

for ( i = 0 ; i < n ; i++)

{

sum = sum + a[i] ;

avg = sum / n ;}

printf("\n Sum of Element of Array is : %.2f ",sum) ;

printf("\n Average of Element of Array are : %2.f ",avg) ;

}

1. Write a Program to display the diagonal elements in a matrix array and also find the sum of them.

Sample input:

1 2 3

4 5 6

7 8 9

Output:

Diagonal Elements are 1 5 9

Sum of diagonal elements = 15

#include<stdio.h>

int main()

{

int array1[10][10],i,j,m,n,sum = 0;

printf("Enter no. of rows :");

scanf("%d", &m);

printf("\nEnter no. of cols :");

scanf("%d",&n);

printf("\nEnter values to the matrix :\n");

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

printf("\nEnter a[%d][%d] value :",i,j);

scanf("%d", &array1[i][j]);

}

}

printf("\nThe Diagonals elements of a matrix are :\n");

if(m==n)

{

for(i=0;i<m;i++)

{

for(j=0;j<n;j++)

{

if(i==j){

printf("%d", array1[i][j]);

sum+=array1[i][j];

}

}

printf("\n");

}

printf("Sum of diagonal elements: %d",sum);

}

else

{

printf("\nMatrix is not a Square Matrix.");

}

return 0;

}

1. Write a Program to find the Maximum and Minimum value in a given array of numbers.

Sample Input:

Enter no. of elements in an array 5

Enter the elements:

1 2 3 4 5

Output:

Maximum of an array 5

Minimum of an array 1

#include<stdio.h>

main(){

int n;

printf("Enter number of elemets in array:\n");

scanf("%d",&n);

int a[n];

int i,j,k,x,l;

for (i=0;i<=n-1;i++)

{

printf("enter element %d:\n",i+1);

scanf("%d",&a[i]);

}

for (j=0;j<=n-1;j++)

{

for (k=j+1;k<=n-1;k++)

{

if (a[j]>a[k])

{

x=a[k];

a[k]=a[j];

a[j]=x;

}

}

}

printf("Maximum Element : %d\n",a[n-1]);

printf("Minimun Element : %d",a[0]);

}

1. Write a Program to find the sum and average of numbers in a matrix format of array

Sample input:

1 2 3

4 5 6

7 8 9

Output:

Sum = 45

Average = 5

#include<stdio.h>

int main()

{

int i,j,m,n;

float a[10][10], sum=0.0, avg;

printf("Enter row and column size:\n");

scanf("%d%d", &m, &n);

printf("Enter matrix elements:\n");

for(i=0;i< m;i++)

{

for(j=0;j< n;j++)

{

printf("a[%d][%d]=",i,j);

scanf("%f", &a[i][j]);

}

}

for(i=0;i< m;i++)

{

for(j=0;j< n;j++)

{

sum = sum + a[i][j];

}

}

avg = sum/(m\*n);

printf("Sum = %.2f\n", sum);

printf("Average = %.2f", avg);

}

1. Write a program in C to add numbers using call by reference

Test Data :  
Input the first number : 5  
Input the second number : 6  
Expected Output :

The sum of 5 and 6 is 11

Test cases:

1. X = 0 , N = 4
2. X = 5 , N = 0
3. X = -3 , N = 3
4. X = 0 , N = 0
5. X = 123, N = 123

#include <stdio.h>

main(){

int num1, num2, result;

printf("\nEnter the two number: ");

scanf("%d %d", &num1, &num2);

result = add(&num1, &num2);

printf("\nAddition of %d and %d is %d", num1, num2, result);

}

int add(int \*no1, int \*no2)

{

int res;

res = \*no1 + \*no2;

return res;

}

1. Write a program in C to store n elements in an array and print the elements using pointer

Test Data :  
Input the number of elements to store in the array :5  
Input 5 number of elements in the array :  
element - 0 : 5  
element - 1 : 7  
element - 2 : 2  
element - 3 : 9  
element - 4 : 8  
Expected Output :

The elements you entered are :

element - 0 : 5

element - 1 : 7

element - 2 : 2

element - 3 : 9

element - 4 : 8

Test cases:

1. N = 16
2. N = -8
3. N = 0
4. N = -10.01
5. N = 11.22

#include <stdio.h>

main() {

int data[20],n,i;

printf("Enter the number of elements :");

scanf("%d",&n);

if(n>0){

for (i = 0; i < n; i++){

printf("Element %d: ",i);

scanf("%d", data + i);

}

printf("You entered: \n");

for (i = 0; i < n; i++)

printf("element %d: %d\n",i, \*(data + i));

}

else{

printf("Invalid input");

}

}

1. Write a program in C to swap elements using call by reference.

Test Data :  
Input the value of 1st element : 5  
Input the value of 2nd element : 6  
Input the value of 3rd element : 7  
Expected Output :

The value before swapping are :

element 1 = 5

element 2 = 6

element 3 = 7

The value after swapping are :

element 1 = 7

element 2 = 5

element 3 = 6

#include <stdio.h>

void swapNumbers(int \*x,int \*y,int \*z);

int main()

{

int e1,e2,e3;

printf("enter the value of 1st element : ");

scanf("%d",&e1);

printf("enter the value of 2nd element : ");

scanf("%d",&e2);

printf("enter the value of 3rd element : ");

scanf("%d",&e3);

printf("\n The value before swapping are :\n");

printf(" element 1 = %d\n element 2 = %d\n element 3 = %d\n",e1,e2,e3);

swapNumbers(&e1,&e2,&e3);

printf("\n The value after swapping are :\n");

printf(" element 1 = %d\n element 2 = %d\n element 3 = %d\n\n",e1,e2,e3);

return 0;

}

void swapNumbers(int \*x,int \*y,int \*z)

{

int tmp;

tmp=\*y;

\*y=\*x;

\*x=\*z;

\*z=tmp;

}

1. Write a program in C to find the factorial of a given number using pointers.

Test Data :  
Input a number : 5  
Expected Output :

The Factorial of 5 is : 120

Test cases:

1. N = 0
2. N = -5
3. N = 1
4. N = M
5. N = %

#include <stdio.h>

void findFact(int,int\*);

int main()

{

int fact;

int num1;

printf("enter a number : ");

scanf("%d",&num1);

findFact(num1,&fact);

printf(" The Factorial of %d is : %d \n\n",num1,fact);

return 0;

}

void findFact(int n,int \*f)

{

int i;

\*f =1;

for(i=1;i<=n;i++)

\*f=\*f\*i;

}

1. Write a program in C to compute the sum of all elements in an array using pointers.    
   Test Data :  
   Input the number of elements to store in the array (max 10) : 5  
   Input 5 number of elements in the array :  
   element - 1 : 2  
   element - 2 : 3  
   element - 3 : 4  
   element - 4 : 5  
   element - 5 : 6  
   Expected Output :

The sum of array is : 20

Test Cases:

1. N = 0,1,3,8,7,-5

1. N = 5,5,5,5,5,5.4
2. N = -2,2,-2,4,-4
3. N = -5,55,30,0.5
4. N = 0.2,2,4,5,8

#include<stdio.h>

int main()

{

int array[23];

int i, sum=0,n;

int \*ptr;

printf("Enter the number of elements:");

scanf("%d",&n);

for(i=0;i<n;i++){

printf("Enter an element:");

scanf("%d",&array[i]);

}

ptr = array;

for(i=0;i<5;i++)

{

sum = sum + \*ptr;

ptr++;

}

printf("Sum : %d",sum);

}

1. Write a program in C to check whether a number is a prime number or not using the function.    
   Test Data :  
   Input a positive number : 5  
   Expected Output :

The number 5 is a prime number.

Test cases:

1. N = P
2. N = 0
3. N = -4
4. N = 11
5. N = 7.2

#include<stdio.h>

int i,ch=0;

prime(int n){

for(i=1;i<=n;i++){

if(n%i==0){

ch+=1;

}

}

return ch;

}

main(){

int n,r;

printf("Enter a number :");

scanf("%d",&n);

if(n>0){

r=prime(n);

if(r==2){

printf("Given number is prime");

}

else{

printf("Given number is not prime");

}

}

else{

printf("Invalid input");

}

}

1. Write a program in C to print all perfect numbers in given range using the function.    
   Test Data :  
   Input lowest search limit of perfect numbers : 1  
   Input lowest search limit of perfect numbers : 100

Expected Output :

The perfect numbers between 1 to 100 are :

* 1. 28

Test cases:

1. 17
2. 26!
3. 143
4. 84.1
5. -963

#include <stdio.h>

int isPerfect(int num);

void printPerfect(int start, int end);

int main()

{

int start, end;

printf("Enter lower limit to print perfect numbers: ");

scanf("%d", &start);

printf("Enter upper limit to print perfect numbers: ");

scanf("%d", &end);

printf("All perfect numbers between %d to %d are: \n", start, end);

printPerfect(start, end);

return 0;

}

int isPerfect(int num)

{

int i, sum;

sum = 0;

for(i=1; i<num; i++)

{

if(num % i == 0)

{

sum += i;

}

}

if(sum == num)

return 1;

else

return 0;

}

void printPerfect(int start, int end)

{

while(start <= end)

{

if(isPerfect(start))

{

printf("%d, ", start);

}

start++;

}

}

1. Write a program to reverse a number using function?(Get the input from user)

Sample Input:

Number: 14567

Sample Output:

Reverse Number: 76541

Test cases:

1. -45721
2. 000
3. AD1947
4. !@#$%
5. 145\*999=144855

#include<stdio.h>

Reverse(int n)

{

int sum=0;

while (n!=0)

{

sum = sum\*10 + n%10;

n /= 10;

}

return sum;

}

main()

{

int rev, num;

printf("Enter a Positive Number: ");

scanf("%d", &num);

if(num>0){

rev = Reverse(num);

printf("The Reverse of given number %d is: %d", num, rev);

}

}

1. Write a C program that uses functions and structures to perform the following operations: i). Addition of two complex numbers ii) subtraction of two complex numbers iii) Multiplication of two complex numbers iv)Division of two complex numbers

Input:

Menu for operation complex numbers

1.Addition 2.Subtraction 3.Multiplication 4.Division

2

Enter First complex number:

"enter real part of complex number: 2

enter Imaginary part of complex number:3

Enter Second complex number:

"enter real part of complex number: 4

enter Imaginary part of complex number:5

Output:

Addition of complex number :

Real part of Subtraction : -2

Imaginary part of Subtraction: -2

Test Cases:

2+5i, a+4i

3+6i. 6+4i

8+4i.12+7i

15-4i. 6-2i

3-8i, 8-6i

1. Write a C program to display the subject and mark information using Dynamic Memory Allocation for Structure

Sample Input:

Enter the number of records: 2

Enter subject 1 and marks:

Science 82

Enter subject 2 and marks:

DSA 73

Sample Output :

Science 82

DSA 73

Test Case:

Enter the number of records :4 (Any details of subject and marks )

Enter the number of records :A

Enter the number of records :1 (CPP 74.5 )

Enter the number of records :1 (CPP seventy)

Enter the number of records :1 (233 75)

1. Write a C program to display the details of student(Name , Age) by passing structures to a function

Sample Input :

Enter No.Students: 1

Enter student 1 Name, Age :AAA, 25

Sample Output:

Student 1 details:

Name: AAA

Age : 25

Test Cases:

No.Student :4 (Any details of student)

No.Student: 5

No.Student: 1( 62, 28)

No.Student: A

No.Student: 1( xxx, 28.2)

1. Write a C program to read contents from the keyboard and write it into the file.

Input:

Enter Name of file to write: Test1.txt

Enter to write:

This is a very good text found while tested.

Output:

Write operation successful!!!

Test Condition:

1. Text.txt
2. Test1
3. Write a C program to merge the two text files.

Input:

Enter file 1: Test1.txt

Enter file 2: Test2.txt

Output:

Merge Successful!!!

The merged Content:

11 22 33 44 55 66 77 88 99

This is a very good text found while tested.

1. Write a C program to read the content from a file and print equivalent ASCII values of each character.

Input:

Content in file:

A E I O U

Output:

65 69 73 79 85

1. Write a C program to delete a file in the system using SYSTEM commands.

Input:

Enter the name of file to be deleted: ram.c

Output:

The file is deleted successfully!!!

The file is not deleted!!! (In case file to be deleted is not available in store.)

.

1. Write C program to Reverse File Contents.

**Input:**

**Enter Name of file to open: file1.txt**

file1.txt has:   
This is line one   
This is line two   
This is line three   
This is line four   
This is line five 

**Output:**   
This is line five   
This is line four   
This is line three   
This is line two   
This is line one

1. Write C program to Copy file contents to another file.

Input:

Enter the filename to open for reading

file3.txt

Enter the filename to open for writing

file1.txt

Output:

Contents copied to file1.txt successfully!!!

1. Write C program to Count the number of words in a line and number of lines in a file.

Input:

Enter file to open: lines.txt

Output:

No. of words in a line = 23

No. of line in file = 6

1. Write a program in C to read the file and store the lines into an array.

Test Data :  
Input the file name to be opened : test.txt  
*Expected Output* :

The content of the file test.txt are :

test line 1

test line 2

test line 3

test line 4

Test cases

Sample.txt

sample1.txt

S123.txt

S@12.txt

Sample S.txt

1. Write a program in C to encrypt a text file. [Go to the editor](https://www.w3resource.com/c-programming-exercises/file-handling/#editorr)

Test Data :  
Input the name of file to encrypt : test.txt  
*Expected Output* :

File test.txt successfully encrypted ..!!

[If you read the file test.txt you will see the following :

������Ʉ�ӄۗ�������ɒ��ђn]

Test cases

Sample.txt

sample1.txt

S123.txt

S@12.txt

Sample S.txt